CHAPTER

5

IMPLEMENTING A PUBLIC KEY INFRASTRUCTURE

After reading this chapter and completing the exercises you will be able to:

- Describe the concepts of public key infrastructure (PKI).
- Describe how PKI can be used to enhance security.
- Understand a PKI hierarchy and trust paths.
- Implement a PKI using Windows 2000 Certificate Server.
- Manage the PKI certificates in Windows 2000.
- Map user accounts to certificates.
- Use a Windows 2000 PKI as part of your security plan.

The previous chapters have described the first steps in designing and implementing a security policy for a large corporation. The focus of the chapters up to this point has been on managing user accounts in Active Directory and controlling the resources users can access based on those user accounts. However, for some large corporations, the focus may need to be broader. Many corporations are multi-national in scope, and may have a variety of working arrangements with subsidiary companies, partner corporations, or clients who use many different technologies to communicate with the corporation from locations throughout the world. For these corporations, the company network is not a closed and isolated LAN used only by people in the office, where the security needs are limited to protecting the network resources from internal attack. Instead, the network is now worldwide and connected with a variety of WAN links, including Internet links. Most corporations deliberately expose a part of their network to the entire world by using Web sites to provide information to clients or potential clients and to sell products and services.

The security plan for a corporation like this must include details on how to secure the network data and services in this distributed environment. The concept of having a single database containing the usernames and passwords for everyone who could ever connect to the network is not a workable solution in this case. Active Directory in Windows 2000 is a very flexible directory system that can be used to extend a network in a variety of ways, but it still depends on one central concept: every user accessing the network has a user account, to which permissions are assigned. As mentioned earlier, Active Directory is based on a shared secret authentication: the user and the Domain Controller both know the password for the user account, and if the user types in the correct password, the user is authenticated and given access to network resources. The limitation of this model is that in a corporation working with a wide variety of other companies and using the Internet to sell products, not everyone connecting to the network has a user account.

Public key infrastructure (PKI) is the most important model currently in place to overcome this limitation. Rather than using a shared secret, PKI is based on users and computers having the right certificates and private and public keys. The certificates and keys can be used to authenticate users and to encrypt all of the data that flows between computers. This chapter introduces the topic of PKI and provides an overview of how the technology works. Most of the chapter deals with the Windows 2000 implementation of PKI, which is Certificate Server. **Certificate Server** is a standards-based PKI implementation that can be used on a worldwide scale, or it can be used only within a corporation.

This chapter introduces PKI and discusses many ways that it can be used in your security plan. However, many of the topics in the remaining chapters in this book will also refer to the certificates and keys that make up a PKI. For example, when setting up virtual private networks (VPN) using IPSec, you may need a PKI to manage the required certificates.

Public Key Infrastructure (PKI) Overview

Public key infrastructure (PKI) is based on three essential concepts: public and private keys, certificates, and Certificate Authorities (CAs).

Public and Private Keys

The basic concept behind PKI is that every user or computer involved in the information exchange has two keys: a private key and a public key. The **private key** is known only to the user who holds the certificate. It can be stored on the computer's hard drive, as part of a roaming profile, or on a different device, such as a smart card. In contrast, the **public key** is made available to anyone who asks for it, either from a directory service like Active Directory or an online certificate authority, or directly from the computer. The private and public keys are related, but there is no way to derive a private key from a public key. These public and private keys are used in a variety of ways.

Data Encryption

The first and most common use for the public and private keys is to encrypt information as it is sent across the network. To understand how the keys are used, consider this example. Suppose that User1 wants to send a message to User2, and the users want to make sure that no one captures and reads that packet as it crosses the network. To insure this, the data must be encrypted while it crosses the network, and only the intended recipient should be able to decrypt and read the message. If the two users are using PKI, User1 uses User2's public key to encrypt the message, and then sends the message to User2. User2 then uses her private key to decrypt the message. The relationship between the public and private keys in this case requires that only the private key can be used to decrypt information that was encrypted with the public key. Therefore, because User2 is the only person with the private key, only User2 can decrypt the message. Anyone capturing this packet on the network does not have the correct private key, and therefore cannot read the message.

In most cases, encrypting an entire message using the public key would be too resource intensive and would create a significant load on the computers. Therefore, the public key is usually used to encrypt only a **session key** (also called a **bulk encryption key** or **symmetric key**), which is then used to encrypt the actual message. The private key is then used to decrypt the session key, and the message is then decrypted using the session key. Figure 5-1 illustrates how the different keys are used to encrypt a message flowing from one user to another.

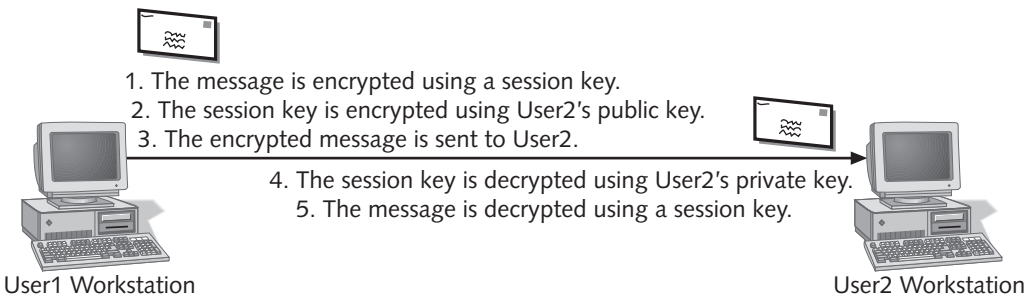

Figure 5-1 Data encryption using private, public, and session keys

Digital Signatures

A second use for public and private keys is to digitally sign and seal messages sent between two users. The purpose of a **digital signature** is to insure the identity of the sender of the message, and also to insure the integrity of the message. In other words, the digital signature insures that the message is actually coming from the person identified as the source of the message, and that the message has not been tampered with on the network. This verification is important in a business environment to protect against someone assuming another person's identity, or to protect against man-in-the-middle attacks where someone substitutes a bogus public key for the original one during the initial public key exchange.

To create a digital signature, a mathematical calculation called a **hash** is applied to the entire message (or to pieces of the message if the message is large). This hash creates a message digest, which is then encrypted using the message sender's private key. The result of the encryption is the digital signature that is sent with the message. When the message recipient gets the message, the same hash is applied to the message, creating a second message digest. Then the sender's public key is used to decrypt the digital signature. If the recipient's message digest is identical to the result of the decrypted signature, then the integrity and authenticity of the message are confirmed. The relationship between the public and private keys in this case is such that when the public key is used to decrypt the message digest, only the digest created by using the private key is accepted. Also, because the digest is created by performing a mathematical function on the entire message, any change to the message also makes the digest invalid, thereby notifying the recipient that the message has been tampered with. Figure 5-2 illustrates the process of digitally signing a message.

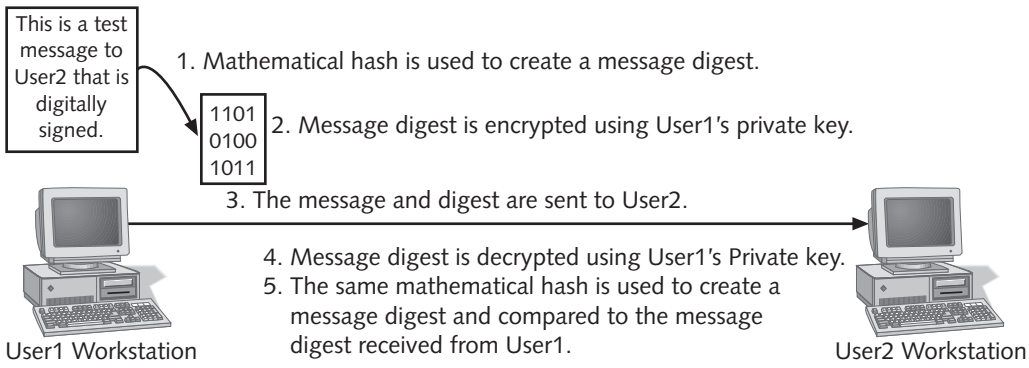

Figure 5-2 Attaching a digital signature to a message

Authentication

PKI also uses certificates and public and private keys to authenticate users or computers. The most common form of authentication is server authentication. For example, a user accessing an e-commerce site that requires the user to enter a credit card number wants to make sure that the server is authentic, or that the server really is an e-commerce server and not an imposter. The authentication in this case is based on a certificate; if the server can prove that it has a valid certificate that the client trusts, then the client assumes that the server is authentic. In some cases, the server is also configured to require client authentication. In this case, the server will ask the client for a certificate and confirm its validity and authenticity. If the certificate is valid, and the client has the private key associated with the certificate, then the client is authenticated.

Regardless of whether the computers are using server authentication, client authentication, or both, the process of performing the authentication is the same. In most cases, the process begins when authentication is requested by one of the computers. For

example, if the server is configured to require client authentication, the server begins by requesting the client authentication. The client computer sends its certificate and the public key to the server. The server then creates a challenge for the client by taking some random information and encrypting it with the client's public key, and then sends that challenge back to the client. The client computer decrypts the challenge using its private key and sends the challenge back to the server. Because the client computer had the right private key to decrypt the challenge, the client is authenticated.

The process of authentication is actually more complicated than this and includes steps such as checking the certificates to make sure that the server is configured to trust the certificates. As well, in most cases, the authentication process is used to create a session key to encrypt all the traffic between the server and client. The details about certificate trusts are included in the section on certificate servers, and the section on Secure Sockets Layer (SSL) describes the process of creating the session key.

Certificates

The second essential component of PKI is the digital certificate. The purpose of a **digital certificate** is to confirm the identity of the certificate holder. When you apply for a certificate from a **Certificate Authority (CA)**, the CA first confirms the identity of the person or company requesting the certificate. When the certificate is granted to the user, the user is also given the associated public key, as well as the private key for the certificate. The certificate is also digitally signed by the certificate authority, which, in effect, adds the certificate authority's stamp of authenticity to the certificate.

The current standard for these certificates is **X.509 Version 3**. The certificate includes information about the person, computer, or service to which the certificate has been issued, information about the certificate itself, and information about the Certificate Authority that issued the certificate. A standard X.509v3 certificate contains at least the following:

- Version
- Serial number
- Signature algorithm ID
- Name of the CA that issued the certificate
- Validity time period
- User name
- The public key information
- The digital signature from the CA

Figure 5-3 shows some of the details of a certificate issued by a Windows 2000 Certificate Server.

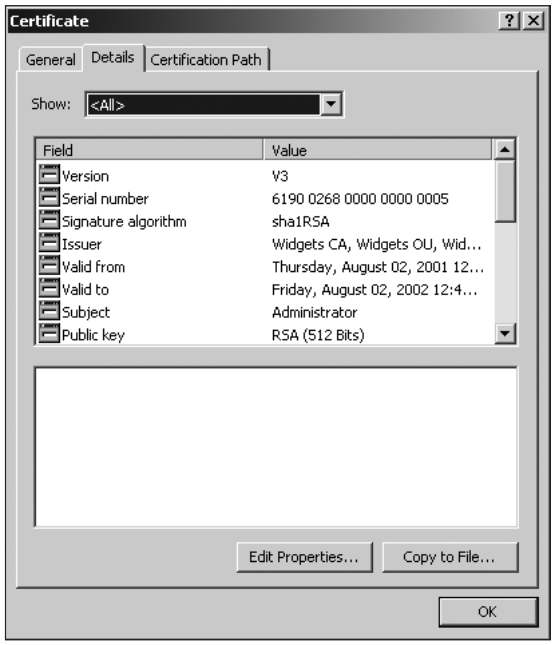

Figure 5-3 The X.509 Version 3 certificate details

Certificate Authorities

The certificates required for PKI are issued by Certificate Authorities. In most cases, CAs are configured in a **hierarchical structure**. A root CA is located at the top of the hierarchy, and subordinate CAs that actually issue the certificates are located under the root. Because of the importance of PKI for the Internet, a wide variety of CAs is currently available, including popular commercial CAs such as Verisign and Thawte. Windows 2000 includes a certificate server component that makes it easy to set up a CA hierarchy for your organization.

The fact that it is easy to set up a CA, and that there are so many CAs available, illustrates an important concept and limitation to PKI. The really important question for you is, "Who issued the certificate to the user?" A user can get a certificate from any one of hundreds of commercial CAs or may even have a certificate from an internal CA in another company. However, you may not want to give that user access to secure information on your network simply because that person has a certificate. The certificate is only as trustworthy as the issuing Certificate Authority. If the certificate has been issued by a trustworthy CA, then you can probably trust the certificate. If the certificate has been issued by a CA that you have never heard of, then you certainly do not want to trust the certificate or the user presenting the certificate.

This verification process is similar to showing a passport to prove your identity when entering a foreign country. When you show your passport, the country you are entering trusts the country that issued the passport. This trust carries with it an assumption that the country that issued the passport has checked to make sure that the person who was issued the passport is the right person, and that the country takes reasonable precautions to insure that its passports are not easily forged. The passport itself can also be examined to insure that it has not been tampered with. As well, the security official will compare what you look like to the picture in the passport to insure that you are the person to whom the passport was issued. In the end, the country that you are entering trusts the country that issued the passport, and if the passport looks like it belongs to you and has not been tampered with, then you are granted access to the country.

When a user presents a certificate to your network in order to gain access to some information, you want to be able to make the same kind of confirmation. First of all, do you trust the CA that issued the certificate? Second, if you do trust the CA, then can you confirm that the certificate has not been tampered with? (Your server will check this by checking the digital signature on the certificate that the client presents.) And third, is the client who is presenting the certificate actually the person to whom the certificate was granted? (The process of authentication confirms that the user has the private key that belongs to the certificate.) If all of these checks come back positive, then you will trust the user and give her access to your network.

The most important and difficult part of this process of accepting the certificate is having to trust the issuing CA. PKI really comes down to whether you want to trust the CA. When a user connects to another computer to establish secure communications, the two sides must agree on a CA that both trust. For example, if User1 wants to establish a secure link to User2, User1's computer will examine User2's certificate and check for the CA that issued the certificate. If the CA is one that User1's computer has been configured to trust, then User1's computer accepts the certificate and begins setting up the secure communication. If User1's computer does not trust the CA, then secure communication fails, or the user is presented with a warning message about the certificate. If you configure your computers to trust the certificates issued by a CA, you are agreeing to accept all valid certificates from that CA. If you are providing access to highly confidential information on the basis of the certificate, then you want to be very sure that the CA checks out all applicants for certificates to insure that only valid users receive certificates.

This process of checking out whether a CA is trusted is built on trust paths. As mentioned earlier, CAs are usually organized in a hierarchical configuration, with a **root CA** and a number of **subordinate CAs** underneath the root. When a root CA is set up, it is configured to be at the top of the CA hierarchy, which means that there is no higher authority. Therefore, the root CA has to issue its own certificate, essentially certifying itself. Figure 5-4 shows an example of this type of certificate.

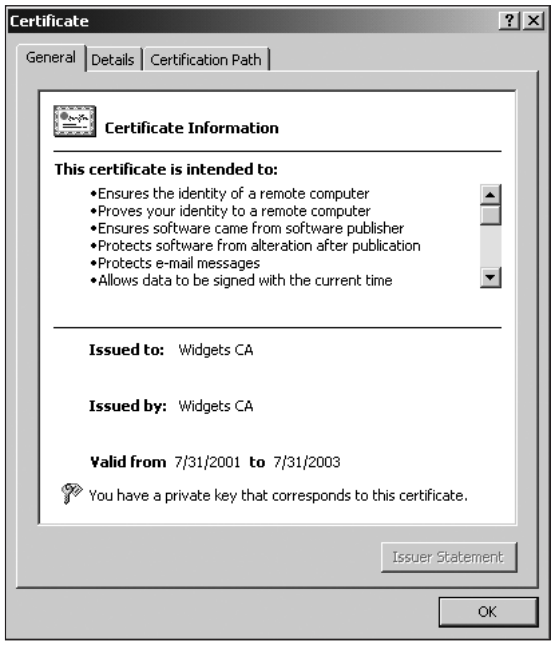

Figure 5-4 A root CA certificate

Subordinate CAs receive a certificate from the root CA, which allows them to hand out certificates to clients. There can also be several levels of subordinate CAs with each subordinate CA receiving its certificate from the CA above it in the hierarchy. This hierarchy forms a **trust path**, which means that all of the certificates handed out by any CAs in the hierarchy share a common trust path. For example, if you have configured your computers to trust a particular root CA, all certificates issued by any subordinate CA to that root CA will be trusted. If two companies have both set up a subordinate CA using the same root CA, the certificates from both companies are automatically trusted in the other company. As long as the certificates belong to the same trust path (they can be traced to a common root CA), the certificates will be trusted. Figure 5-5 illustrates how a trust path works.

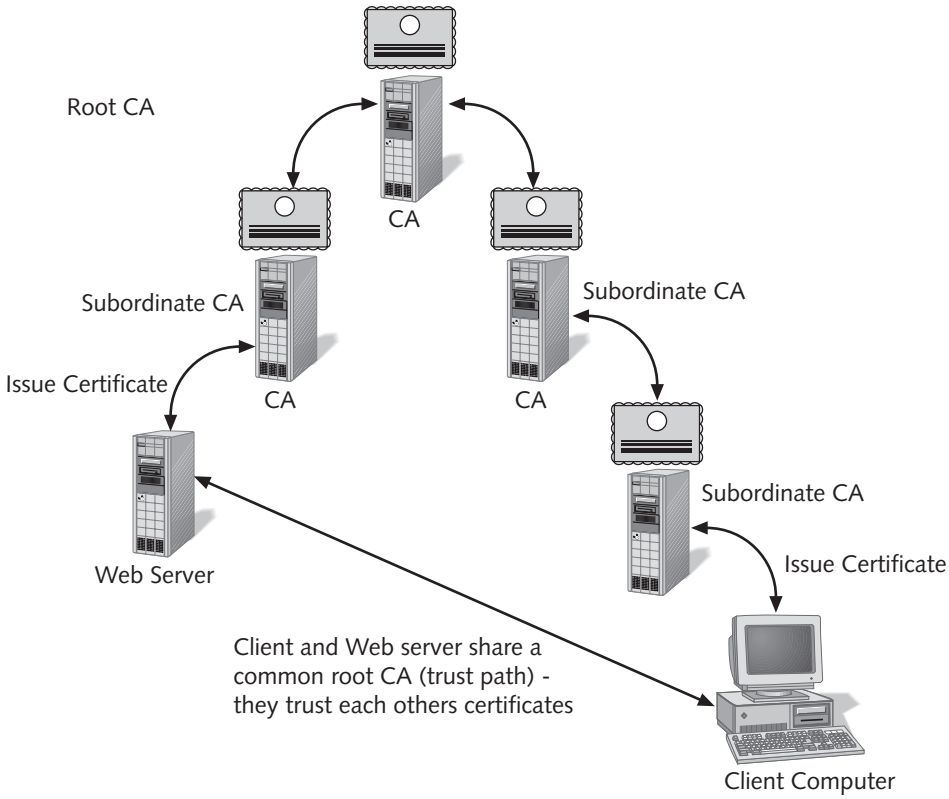

Figure 5-5 Trust paths in PKI

Application Support

The goal of PKI is to provide a secure environment for applications, computers, and users to exchange information. PKI is a service; it is useful only when it is actually implemented by an application on the network. This is an important point as you look at implementing PKI as part of your security plan. What applications are using the enhanced security? And which users are using those applications?

Currently, several applications are taking advantage of PKI technology:

- Secure Web sites
- Secure e-mail
- Smart card logon
- Digitally signed content
- Encrypted File System (EFS)
- IP Security (IPSec)

As more applications are used on a distributed basis, this list will grow.

Secure Web Sites

One of the primary motivations for the adoption of PKI is the desire to have secure Web sites. Three components are needed to have an entirely secure Web site:

- **Server authentication**—The client must verify the server's identity.
- **Client authentication**—The server must verify the client's identity, and then apply the permissions given to this user.
- **Confidentiality**—All data between the client and server must be encrypted.

Most secure Web sites depend on **Secure Sockets Layer (SSL)** or **Transport Layer Security (TLS)** to meet these requirements. Both SSL and TLS require PKI.

To understand how SSL works, consider the following example where a client is accessing a secure Web site and trying to get access to confidential data. The Web site has been configured to require all three components to a secure Web site. The process of authenticating the client and server, and creating the session key to encrypt all of the data sent between the two servers, is called the **SSL handshake**. Figure 5-6 and the following steps describe the SSL handshake.

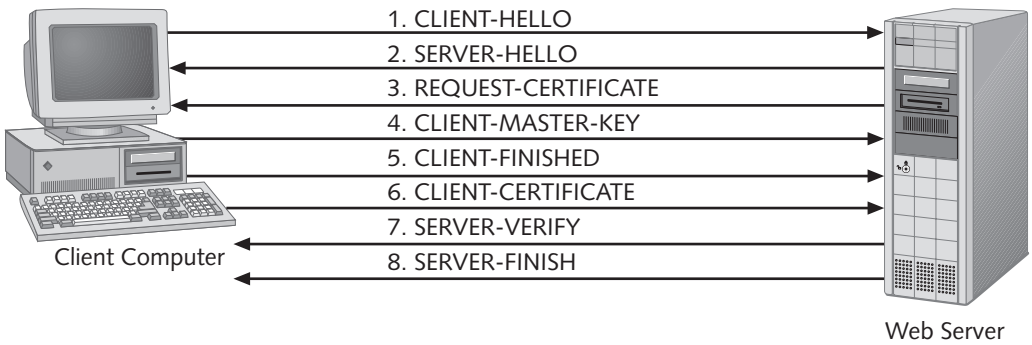

Figure 5-6 The SSl handshake

1. The client initiates the handshake by sending a CLIENT-HELLO packet to the server. This packet includes a challenge, which is randomly generated data and a listing of the types of cryptography the client is capable of supporting.
2. The server responds by sending a SERVER-HELLO message, which includes a connection identifier, the server's certificate, including the CA's digital signature and the server's public key, and a listing of the types of cryptography supported by the server.
3. If the server is configured to require or request client authentication, the server also sends a REQUEST-CERTIFICATE, which includes a challenge and the type of cryptography required.

4. The client responds to the SERVER-HELLO message by choosing the highest level of cryptography supported by both client and server, and then sending a CLIENT-MASTER-KEY message. This message includes a **session key** that the client will use to encrypt all data sent to the server. The session key is encrypted with the server's public key.
5. The client then sends a CLIENT-FINISHED key, which includes a connection identifier and is encrypted using the session key.
6. The client also responds to the REQUEST-CERTIFICATE with a CLIENT-CERTIFICATE, which includes the client certificate.
7. The server decrypts the CLIENT-MASTER-KEY message from the client and replies with a SERVER-VERIFY message. This message includes decrypted information from the client message, thus proving that the server has the private key for the certificate that it sent to the client.
8. The server responds to the CLIENT-CERTIFICATE message with a SERVER-FINISH message.

At this point the client and the server have both been authenticated, and they have agreed on a session key that will be used to encrypt all of the information between the two computers. In some cases, the user may still need to enter a user name and password to authenticate to the Web server, but now all of the information, including the user name and password will be encrypted.

Most secure Web sites are not configured to require client authentication. A company may be hoping to attract millions of users to their Web site, and managing this number of certificates is not practical. Instead, the Web server has a certificate assigned to it that the clients use to authenticate the server and negotiate the session key. In most cases, the Web server's certificate comes from a commercial CA such as Verisign because most Internet browsers are configured by default to trust these certificates.

Secure E-mail

Another important application using PKI is secure e-mail. Many mail servers, such as Domino Server/Lotus Notes and Microsoft Exchange Server, support secure e-mail. Secure e-mail provides two options:

Digital signatures—The digital signature insures the origin of the mail message, as well as confirming that the message has not been modified. The message is signed with the sender's private key, and the signature is checked with the sender's public key.

Data encryption—Data encryption is used to insure that the content of the mail message cannot be read by any user other than the holder of the private key. The message is encrypted using the recipient's public key and decrypted using the recipient's private key.

One of the disadvantages of using secure e-mail is the difficulty in sending mail between organizations. Secure e-mail does not provide the same functionality for public key

exchange that accessing a Web site does. For example, if you want to send someone an encrypted e-mail, you need that person's public key. If the other person works for a different organization, you will probably not be able to access the public key, unless you have arranged some type of certificate exchange. One of the options in dealing with this issue is to use a third-party package that encrypts and signs e-mail between two users using the same third-party software. **Pretty Good Privacy (PGP)** provides this type of technology and maintains a Web site where the certificates for all of its subscribers are accessible. Another option is to use a technology that is based on an open standard. **Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3** is a proposed standard for secure e-mail, and as long as both e-mail systems follow the S/MIME standards and share a common CA, they can exchange secure e-mail.

Smart Card Logon

The use of smart cards is an example of how Active Directory and PKI can be used together to provide additional authentication options and enhance security within a network. A **smart card** is a credit-card-sized device that stores the user's certificates and private keys. The certificates are written into the erasable, programmable memory (EPROM) on the card. Because all of the users of a smart card are likely to be employees of the company, rather than external clients, you can quite easily set up a Windows-2000-based PKI and use it to issue the certificates needed for the smart cards.

To use a smart card, the user inserts the card into any computer with a card reader. The Windows 2000 implementation of the smart card interprets the insertion as being the same as [CTRL+ALT+DEL] and prompts the user to enter a PIN number. The authentication process now uses Kerberos to begin the authentication. The KDC first checks the certificate's authenticity and insures that the issuing CA is trusted, and then uses the user information from the certificate to look up the user in AD and complete the ticket-granting process. The smart card logon enhances the security of the network because there is no password to be guessed, and both the physical device and the PIN number are needed to authenticate the user. The other important advantage of the smart card is that users carry their certificates with them. Therefore, they can log on at any computer that has a smart card reader in the network.

Digitally Signed Content

Another Internet-related activity where PKI is useful is in verifying content that has been downloaded from the Internet. Users may download what appears to be an innocent application, ActiveX control, or Java applet from the Internet, only to find that it contains malicious code that causes damage to their systems. In some cases, the developers of the application may have deliberately designed the application with the malicious code; in other cases, the attacker may have modified a legitimate application.

Microsoft's solution to this problem is **Authenticode**, allowing software developers to digitally sign any content on the Internet. The software developers must apply to a CA to get a certificate that allows them to sign the downloadable content with a private key.

When the client downloads the content from the Internet, the computer confirms that it trusts the CA that issued the certificate. The computer then uses the developer's public key to confirm the signature, thus ensuring the identity of the content provider, as well as ensuring that the content has not been altered. By default, Windows 2000 checks the digital signature of any content that has been downloaded from the Internet before it allows the installation of the application. If the computer is not configured to trust the CA, or if the digital signature is not valid, the user is warned before the installation starts.

Encrypted File System (EFS)

As discussed in Chapter 3, "Securing Resources on Windows 2000 Servers," Windows 2000 allows users to encrypt files that are saved on a hard drive so that the files are secure from all users except the user with the correct private key. When a file is encrypted, file encryption keys (similar to a session or bulk encryption key) are used to encrypt the data, and then the file encryption keys are encrypted using the user's public key. The encrypted keys are then stored with the file in the file header. To decrypt the file, the file encryption keys are decrypted using the user's private key. Only the user whose public key was used to encrypt the keys has the right private key. Once the file encryption keys have been decrypted using the private key, the data is decrypted.

In case the private key is lost, Windows 2000 provides a recovery policy that always makes it possible for the data to be recovered. By default, the administrator of a standalone machine and the administrator account on the first DC created in a domain are the recovery agents for the network. This means that these accounts also have a certificate issued to them that includes a private key that can be used to decrypt any encrypted file on the computer.

EFS uses public key technology but does not necessarily require a CA to enable EFS. Any Windows 2000 computer can create a certificate for a user the first time a file is encrypted on that computer. In a large company, however, you will probably want to set up a PKI using Windows 2000 Certificate Server to provide more flexibility in managing the certificates needed for data recovery, as well as allow for roaming users.

IP Security (IPSec)

IP Security (IPSec) is an Internet Engineering Task Force (IETF) solution that can use PKI to protect data on the network. With IPSec, all data transmitted on a network can be encrypted. The encryption takes place at the Internet Protocol (IP) network layer, which means that the encryption is invisible to both the applications at higher levels of the TCP/IP stack and the lower physical layers. Applications do not need to be IPSec-aware, because when the data is passed to the application, it is no longer encrypted. At the lower physical layers, the data is encrypted, but these layers see the data only as a packet that must be delivered to a destination without any need to see the contents of the packet.

IPSec can use either a shared secret technology (the user must have an account in the directory and the password is the shared secret), or the public and private keys to encrypt and decrypt the data. In either case, the two computers use the exchange of information (either the shared secret or keys) to determine what level of encryption is supported by both computers and to negotiate a session key. All data is then encrypted using a session key.

IPSec is the encryption mechanism used in **Layer 2 Transport Protocol** (L2TP), an emerging technology for creating Virtual Private Networks (VPNs) throughout the Internet. This technology is fully enabled in Windows 2000, but the lack of complete standards makes interoperability with other networks difficult. L2TP is based on open standards, and as the standards develop, the ability to connect a variety of systems using IPSec increases.

IPSec is covered in more detail in Chapter 7, "Securing Network Communications."

Planning and Implementing a Public Key Infrastructure

With this basic understanding of PKI, you are ready to think about how PKI will fit into your security plan. Relatively few large corporations have developed a comprehensive implementation of PKI at this point, but the increased need for distributed security and the requirement to protect data on the Internet means that many more corporations are looking at PKI. Windows 2000 can be used to implement a complete PKI solution in a corporation. By installing Certificate Server, you can manage all the certificates and keys needed to use any of the public-key-enabled applications for the users in the company, as well as for any other users to whom you want to issue certificates.

If your corporation is thinking about using the Windows 2000 implementation of PKI, your security plan will include the details on Certificate Server installation and configuration. Some of the issues to include in the planning are:

- Designing the Certificate Server hierarchy
- Planning the Certificate Server type
- Identifying client certificate needs
- Defining certificate policies
- Planning for certificate revocation

Designing the Certificate Server Hierarchy

Certificate Authorities (CAs) are usually arranged in a hierarchical configuration, with a root CA at the top and subordinate CAs below the root. The root CA issues its own certificate; in other words, it certifies itself because there is no CA above it. The root CA

gives the subordinate CA a certificate. This allows the subordinate to hand out certificates to clients or to other subordinate CAs.

The CA hierarchy can include as many levels as necessary, but one of the most common configurations includes at least three levels, as described in Table 5-1.

Table 5-1 CA hierarchy levels

CA Level	Level Description
Root CA	At the top of the hierarchy is the root CA. The only certificates that the root CA issues in this configuration would be certificates for the next level of CAs—the intermediate CAs.
Intermediate CA	The intermediate CA is configured to hand out certificates to the next level of CAs, which are the issuing CAs.
Issuing CA	Issuing CAs are the CAs that actually assign certificates to clients. For a smaller company, a two-level structure can be used where the intermediate CA is also the issuing CA.

There are a number of reasons why this configuration is used:

- *Isolation and protection of the root CA*—In most cases, the root CA is not connected to the network, but is isolated in a locked room. The reason for this is to guard against security risks if anyone compromises the root CA. If an attacker can get access to the root CA, he is able to create certificates for any subordinate CAs. This attack would compromise the entire trust path because of the potential for illegitimate subordinate CAs. In some cases, the subordinate CAs are also kept offline.
- *Flexibility*—Using a hierarchical structure allows for increased flexibility, especially at the issuing CA level. If your company is using Active Directory, each OU could be assigned a particular CA, or a different CA could be used to assign each type of certificate.
- *Scalability*—This model can be scaled to almost any size and still retain a single root at the top simply by assigning more intermediate and issuing CAs.
- *Trust paths*—All of the CAs that form the hierarchy under a root CA trust each other because they all trust a common root. When a certificate issued by a CA in one part of the organization is trusted in any part of the organization, no further administration is required.

Windows 2000 Certificate Server can be used at any or all of the levels in the hierarchy. If an organization is concerned primarily with the internal use of certificates, the first Certificate Server can be configured as a root CA, and all subordinate CAs can be installed using a certificate from the root CA. However, the first Certificate Server in the organization could also be installed as a subordinate CA to one of the commercial root CAs on the Internet (provided by organizations like Verisign). To do this, the organization must apply to the commercial CA to receive a root certificate. The

Certificate Server can be used to create a certificate request, which is then forwarded to the CA, along with the required company information. The CA determines whether the request meets the criteria to receive a certificate. (The level of criteria verification required depends on the level of certificate requested.) If the request meets the criteria, the CA uses its private key to digitally sign the certificate and forward it back to the requesting CA. The certificate is then installed on the subordinate Windows 2000 CA.

This process creates a trust path to the commercial CA. This means that certificates issued by the organization's CA are now trusted by any other organization that also has a trust path to the same commercial CA. For example, MiniWidgets and MegaWidgets could both establish a CA that was subordinate to a root CA at Verisign. If a user from MiniWidgets were issued a certificate by the company CA, the user would be able to set up an SSL connection to secure the web server at MegaWidgets, because both CAs trust the same root.

Windows 2000 Certificate Server also provides an alternative to using a common root domain if two companies want to trust each other's certificates. This alternative is called a cross certificate. The **cross certificate** is used when a trust is established between two root domains without sharing a common root. To configure a cross certificate, you would request a certificate from a CA in the other organization, and then install the certificate on one of your Certificate Servers. This cross certificate can be configured at any level in the hierarchy or even at a client level.

Planning the Certificate Server Type

Once you have designed your CA hierarchy, the next step is to decide what type of Certificate Server you need to install. You have four choices when you install Certificate Server: enterprise root CA, subordinate enterprise CA, standalone root CA, and subordinate standalone CA.

Enterprise Root CA

The first option when installing a Certificate Server is to install an enterprise root CA. If you are managing certificates only for users in your organization, and all of the users that require certificates have Active Directory accounts, then the **enterprise CA** hierarchy is the best option. The enterprise CA requires Active Directory and is completely integrated with it. This integration can simplify the administration of certificates, because you can configure policies that automate the process of granting, renewing, and revoking certificates. When you install an enterprise CA, several components are automatically configured:

- The root CA is added automatically as a trusted root for all users and computer certificates in the domain.
- All certificates are automatically approved or denied based on the policies set for the domain or OU. The certificate request is never set to pending.

- The user information that would normally be required during a certificate request is automatically extracted from Active Directory.
- Certificates for authentication using smart cards can be issued.
- The **certificate revocation lists (CRLs)** and client certificates are published to Active Directory.
- Certificates can be automatically mapped to user accounts.

Since this integration with Active Directory automates many of the components of certificate management, your first choice when installing Certificate Server should be an Enterprise Root CA. If you are creating a CA hierarchy, then the enterprise root CA is usually configured to provide certificates only to subordinate enterprise CAs.

Subordinate Enterprise CA

The subordinate enterprise CA is certified by the enterprise root CA and, in most cases, is responsible for handing out certificates to the clients in the domain.

Standalone Root CA

If you are going to be handing out certificates to users outside your organization, or if you want to install a subordinate CA to a commercial CA, then you need to install a **standalone CA**. Standalone CAs do not require Active Directory, but can make use of Active Directory, if available. Because the standalone CA is not as tightly integrated with Active Directory, many of the automated certificate management tools are not available. Some characteristics of the standalone CA are as follows:

- Since the server does not have access to Active Directory, the user applying for a certificate must provide all the needed personal information.
- All certificate requests are set automatically to pending on the certificate server and must be manually accepted by the certificate administrator.
- Smart card certificates for network authentication cannot be issued, although other smart card certificates can be used.
- If the standalone CA is configured to use Active Directory, the standalone CA is added automatically as a trusted root for all users and computer certificates in the domain. The client certificates and certificate revocation lists (CRLs) are also published to Active Directory.

If you are installing only one CA, and you need to assign certificates to many users outside the organization, you should install a standalone root CA. If you are creating a CA hierarchy, then the standalone root CA should be configured to provide certificates only to subordinate standalone CAs.

Subordinate Standalone CA

The subordinate standalone CA is certified by the standalone root CA and is, in most cases, responsible for handing out certificates to clients. After the root CA of either type

has been created, the number of levels in the hierarchy can be defined by installing multiple subordinate CAs, with each level certifying the level below it.

Identifying Client Certificate Needs

Once you have defined the CA hierarchy and certificate server type, the next step in designing the PKI is identifying the clients' certificate needs. Essentially, this process consists of answering three questions:

- What applications require certificates? In many companies, the only certificate requirement is to secure a Web server. If this is the case, then you need only to implement the certificates that provide for user and server authentication. Other companies might be deploying a broad range of applications that require PKI and require several other certificate types.
- Which users must use certificates? Do all the users in the organization need certificates? In many companies, the only people who require certificates are users who spend significant time outside the office and need to be able to secure communication with the office.

Because certificates can be managed using group policies, grouping users that have similar certificate requirements into common OUs can make the management of certificates easier.

- Do the users that require certificates all have Active Directory accounts, or are many of the users outside your organization? If all of the certificate users have Active Directory accounts, then management of the certificates can be largely automated by using an enterprise CA. If many certificate users do not have Active Directory accounts, you will have to spend more time manually administering the certificates.

Defining Certificate Policies

The last part of planning the PKI is defining the certificate policies for the organization. There are a number of decisions that must be made:

- How are certificates assigned? If you are using the enterprise CA, every user and computer can be assigned certificates automatically when the user or computer object is created in Active Directory. Or you can require each user to manually request a certificate, using either the certificates snap-in or the Web interface.
- Do you use group policies to implement certificate policies?
- What are the cryptographic and key length requirements for the certificates?
- How long are certificates valid?
- How are certificates renewed?

- How is the process of revoking certificates managed? How is the Certificate Revocation List (CRL) made available to all subordinate CAs and clients?
- If you are using smart cards, what is the enrollment process to assign users the required cards and certificates?

After you have decided what certificates you need to assign to the clients, you may have to configure the certificate server to support all of the certificates. Figure 5-7 shows the types of certificates that a Windows 2000 Certificate Server can assign by default. You can add additional types of certificates based on the built-in certificate templates, or you can create a custom template for your own use. Table 5-2 lists the certificate templates available in Windows 2000 Certificate Server.

Figure 5-7 The default certificates available in Windows 2000 Certificate Server

Table 5-2 The certificate templates available in Windows 2000

Certificate Type	Purpose for the Certificate
Administrator	Used for authenticating clients and for EFS, secure mail, certificate trust list (CTL) signing, and code signing.
Authenticated Session	Used for authenticating clients.
Basic EFS	Used for EFS operations.
CEP Encryption (offline request)	Used to enroll Cisco routers for IPSec authentication certificates from a Windows 2000 CA.
Code Signing	Used for code signing operations.
Computer	Used for authenticating clients and servers.
Domain Controller	Used for authenticating Domain Controllers. When an enterprise CA is installed, this certificate type is installed automatically on Domain Controllers.
EFS Recovery Agent	Used by the EFS encrypted-data recovery agent.
Enrollment Agent	Used for authenticating administrators that request certificates on behalf of smart card users.
Enrollment Agent (computer)	Used for authenticating services that request certificates on behalf of other computers.

Table 5-2 The certificate templates available in Windows 2000 (continued)

Certificate Type	Purpose for the Certificate
Exchange Enrollment Agent (offline request)	Used for authenticating Exchange Server administrators that request certificates on behalf of secure mail users.
Exchange Signature Only (offline request)	Used by Exchange Servers for client authentication and secure mail (used for signing only).
Exchange User (offline request)	Used by Exchange Servers for client authentication and secure mail (S/MIME).
IPSec	Used for IPSec authentication.
IPSec (offline request)	Used for IPSec authentication.
Root Certification Authority	Used for root CA installation operations. The root CA assigns this certificate to itself when you install a root CA.
Router (offline request)	Used for authentication of routers.
Smart Card Logon	Used for client authentication and logging on with a smart card.
Smart Card User	Used for client authentication, secure e-mail, and logging on with a smart card.
Subordinate Certification Authority (offline request)	Used to issue certificates for subordinate CAs.
Trust List Signing	Used to sign CTLs.

Windows 2000 Certificate Server Implementation

Once you have completed the planning phase for your implementation, you are ready to start installing and configuring certificate servers. The process in implementing and managing Certificate Server involves a number of steps:

- Installing the root and subordinate servers
- Configuring servers and applications to use certificates
- Managing user certification requests
- Managing certification revocations
- Managing group policy settings to manage certificates
- Mapping user accounts to certificates
- Renewing certificates

Installing Certificate Servers

The first Certificate Server that must be installed if your organization is managing its own root CA is either an enterprise or standalone root CA. The procedure for installing either type of root CA is almost identical. The following procedure describes how to install an enterprise root CA.

1. You must be logged in as a member of the local Administrators account if you are installing a standalone root CA, and as a member of the Domain Admins group if you are installing an enterprise root CA.
2. Click **Start**, point to **Settings**, then click **Control Panel**. The control Panel window opens.
3. Double-click the **Add/Remove Programs** icon. The Add/Remove Programs dialog box opens.
4. Click the **Add/Remove Windows Components** tab.
5. Select **Certificate Services**. You are warned that the computer cannot be renamed, added, or removed from a domain once this service is installed.
6. Click **Yes** to accept the warning message, and then click **Next**. You are presented with the choice of what type of CA you are installing. See Figure 5-8.

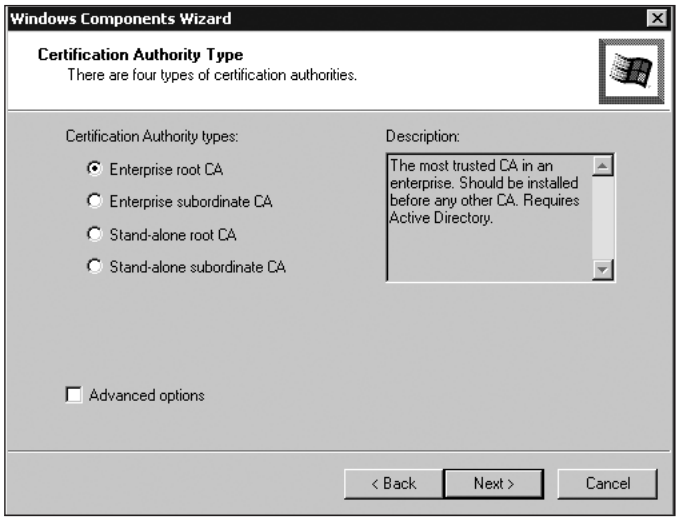

Figure 5-8 Choosing a Certificate Server type

7. Verify that **Enterprise Root CA** is selected, and then click **Next**.
8. Fill in the required company information as shown in Figure 5-9 and then click **Next**.

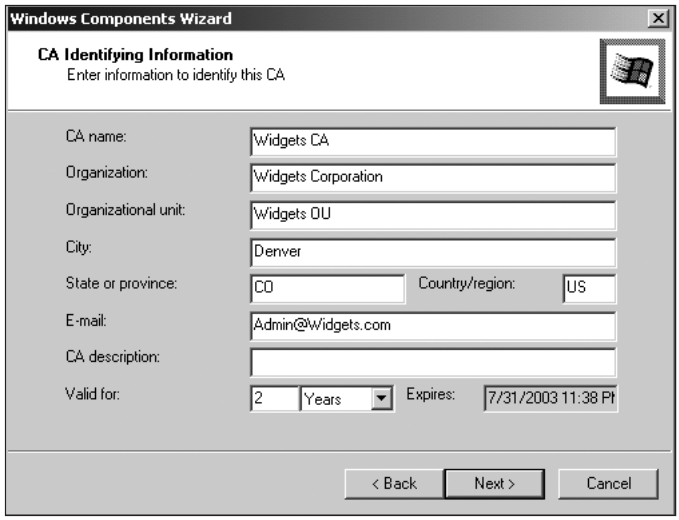

Figure 5-9 Configuring the company information when installing a root CA

9. Click **Next** to accept the default Data Storage Location.
10. If necessary, click **OK** to accept that Internet Information Services will be stopped on your computer.
11. You may be prompted for the Windows 2000 Server CD. Insert the CD-ROM and then click **Enter**.
12. Click **Finish**.

The procedure for installing a subordinate CA is similar to the procedure outlined above. The primary difference is that you have to request a certificate from a higher level CA. Figure 5-10 shows the dialog box you will see during the installation. You have a choice of sending the request to an online CA, or preparing a text file request. If you have implemented a CA hierarchy where the root CA is offline, you have to save the request to a file, and then take the request to the root CA to fulfill it.

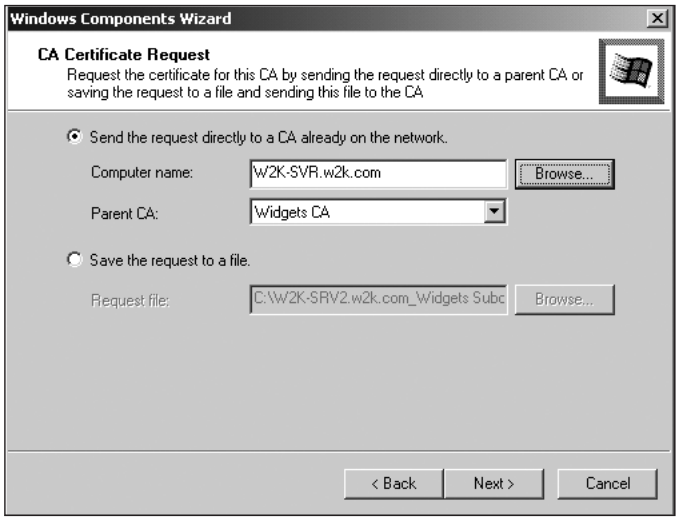

Figure 5-10 Configuring a certificate request for a subordinate CA

Once you have configured the Certificate Servers, clients can begin to connect to the server to obtain certificates. The procedures the clients will use to obtain the certificates will be described in the section on Certificate Server Client Implementation.

Configuring Servers to Use Certificates

After installing the Certificate Server, the next step is to configure the services and applications on the network to use certificates. The procedure for configuring each application is different. For example, if you want to use certificates to secure e-mail by using S/MIME on a Microsoft Exchange Server, you have to install a Key Management server in Exchange, and then apply for certificates using the Key Management administrative tool. If you want to use certificates to create IPSec tunnels through the Internet, you will have to configure the certificate use when you configure IPSec.

The most common way that certificates are used is to secure Web servers. The following procedure describes how to install a certificate on an Internet Information Server.

1. Click **Start**, point to **Programs**, point to **Administrative Tools**, then click **Internet Service Manager**.
2. Expand *yourcomputername*.
3. Right-click the Web site that you want to configure to use the certificate, and then click **Properties**.
4. Click the **Directory Security** tab. Figure 5-11 shows the interface.

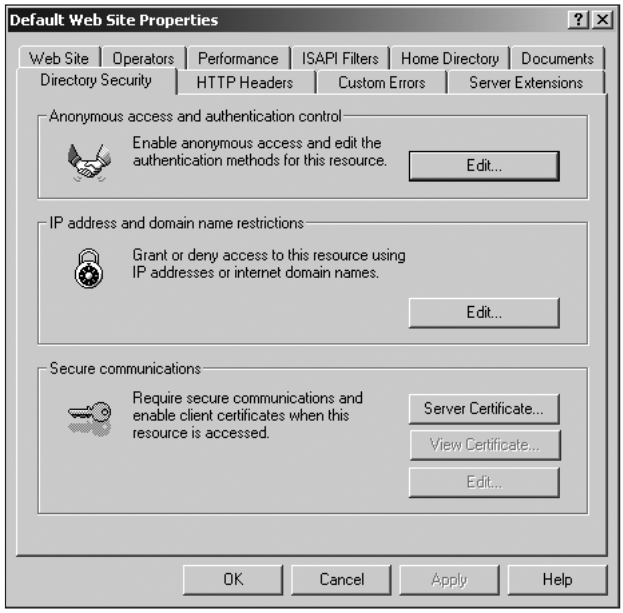

Figure 5-11 The Directory Security tab for the default Web site

5. Click **Server Certificate**. The **Welcome To The Web Server Certificate** wizard starts. Click **Next**.
6. You are presented with a choice of three methods for acquiring the server certificate. See Figure 5-12. If you already have a certificate, click **Assign an existing certificate**. If you want to restore a backup certificate stored in Key Manager, click **Import a certificate from a Key Manager backup file**. If you do not have a certificate, click **Create a new certificate**. Click **Next**.

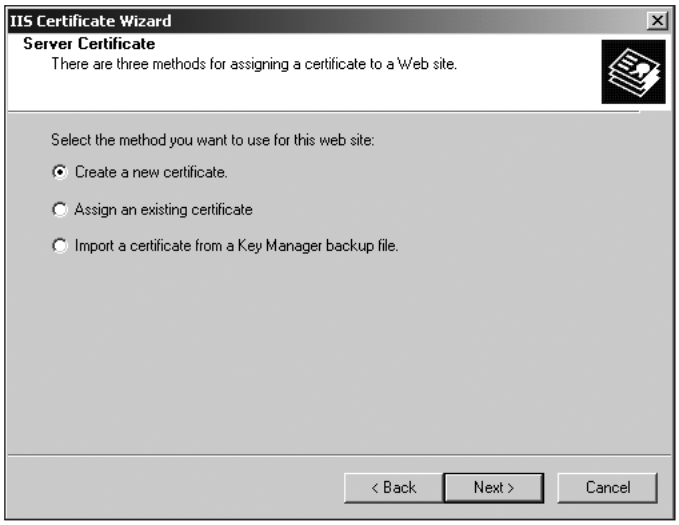

Figure 5-12 Choosing the method for assigning a certificate

7. You are presented with a choice of how you want to apply for the certificate. See Figure 5-13. If you want to use a certificate from a commercial CA, click **Prepare the request now, but send it later**. You will then be presented with several dialog boxes that require information about the Web site that you are securing, as well as some corporate information. When you have finished, the Wizard creates a text file that contains a key that you will send to the certificate authority to obtain the certificate. In most cases, you are also required to provide some other corporate information (such as a tax number or articles of incorporation) that the CA can use to establish your company's identity.

   If you are using your own CA to create the certificate, and the CA is accessible on the network, then click **Send the request immediately to an online certificate authority**. Click **Next**.

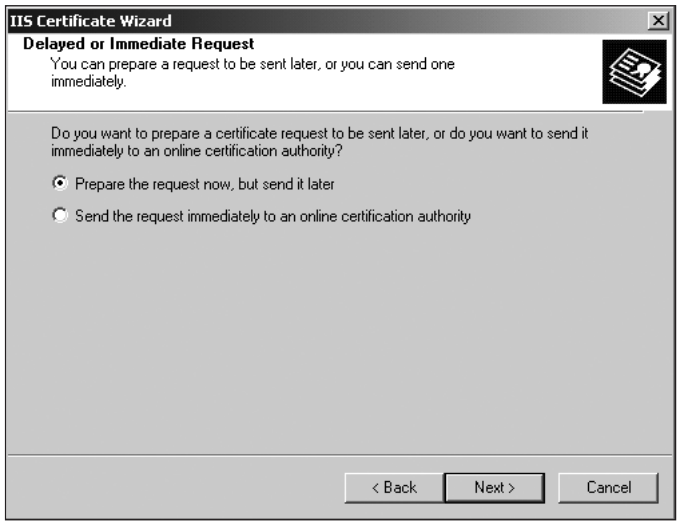

Figure 5-13 Choosing how to apply for the certificate

8. Choose the name of the Web site to which you are assigning the certificate, and choose the length of the encryption key that you want to use. In most cases, you should choose 1024 bits. See Figure 5-14. Click **Next**.

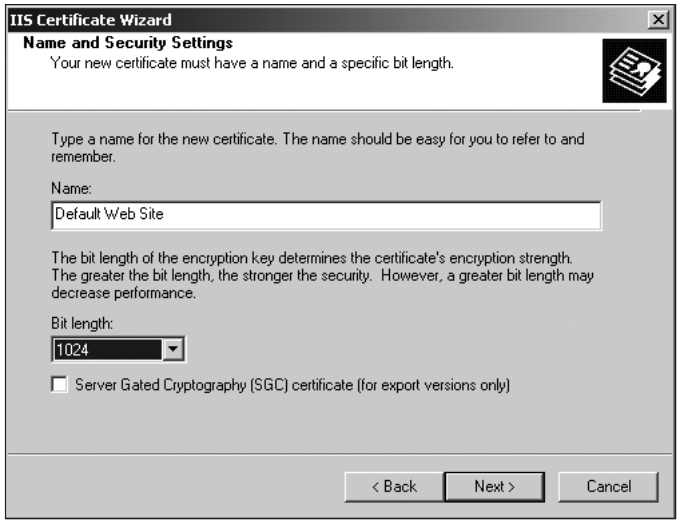

Figure 5-14 Configuring the web site and encryption key

9. Type the information about the organization and the organization unit. Click **Next**.

10. Type the common name for the Web site that you are configuring for SSL. You must type in the name that other computers will be using to access this Web site. If the server is accessible from the Internet, you must type in the DNS name for the Web site. If you do not use the correct name, clients will get error messages when connecting to the server. Click **Next**.
11. Enter the information about which online CA you want to send the request to. The web server will try to locate any online CAs. If you are using an Enterprise CA and the Web server is a member in an Active Directory domain, the Enterprise CA will be pre-selected. Click **Next**.
12. Review the certificate request information. Click **Next**.
13. If the online CA is accessible, the certificate request is sent to the CA. If the CA is configured to grant this type of CA, then the certificate will be granted and installed on the Web server. If the CA is not configured to automatically grant the certificates, the certificate request is queued on the CA until the CA administrator grants the request. Then the certificate can be installed on the local server. Click **Finish**.

    This procedure has installed the certificate on the Web server, and clients will now be able to connect to the server using SSL. However, if you want to force clients to use SSL, the Web server requires further configuration.
14. To configure the server to require SSL, click the **Directory Security** tab of the Web site that you are configuring and then click **Edit** in the Server Communications section. See Figure 5-15. You have a number of configuration options in this dialog box. Table 5-3 provides the details.

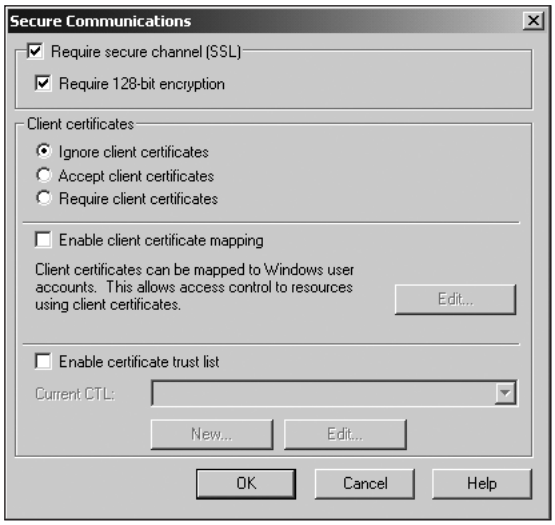

Figure 5-15 Configuring a Web server to require SSL

Table 5-3 Secure communications configuration options

Configuration Option	Explanation
Require secure channel (SSL)	Choosing this option forces all clients to use SSL when they connect to the server. If the client does not type https:// when accessing the server, they will get an error message.
Require 128-bit encryption	Choosing this option means that only clients that support 128-bit encryption will be able to connect to the server. If you do not check this option, 40-bit encryption is used.
Client Certificates	Use this option to configure whether or not you want to require client certificates when clients connect to the server. If you choose Ignore client certificates, the server does not request a certificate from the client. Choosing Accept client certificates means that the server will request a certificate, but accept client connections without a certificate. Choosing Require client certificates means that the server will accept connections only from clients that have certificates.
Enable client certificate mapping	Use this option to configure a certificate to user account mapping. If you choose this option, you can link Active Directory user accounts to specific certificates. When a user connects to the server using the certificate, he will be given the level of permissions granted to the user account.
Enable certificate trust list	Use this option to configure approved CAs for this Web site.

15. Accept the changed security on the virtual directories under the Web site. Click **OK**.
16. Right-click **Default Web Site**. A shortcut menu opens. Click **Stop**.
17. Right-click **Default Web Site** again, and then click **Start**.

Managing Certification Requests

As clients begin to request certificates, you need to manage the requests. The easiest way to manage certification requests is to use an enterprise CA and configure group policies to automatically approve the certificate requests that you want approved. However, if you are using a standalone CA, the default configuration means that all certificate requests are queued on the certificate server as a pending request. You must manually approve the certificate requests.

All of the Certificate Server administration is performed using the Certificate Authority MMC snap-in. This snap-in is added automatically to the Administrative Tools folder when Certificate Server is installed.

To approve pending client requests, use the following procedure.

1. Open the Certification Authority console.
2. Expand the Pending Requests folder and examine the certificate request information.
3. Right-click the certificate request and choose **Issue** or **Deny**.
4. The client must then install the certificate on his/her computer.

You can change the default configuration on a standalone CA to always approve certificate requests by opening the Properties of the CA in the Certification Authority console and choosing the Policy Module. In most cases, you should not configure the standalone CA to approve all certificate requests automatically, because then you have little control over who gets a certificate. Most companies will accept the default setting and extra administrative effort of requiring approval of all certificates.

Managing Certification Revocations

An essential part of managing Certification Services is to manage the revocation of certificates. If an employee leaves the company, or if a temporary working relationship with a business partner who had a certificate has ended, you must revoke these certificates to insure that they are not used inappropriately.

When a certificate is revoked, it is added to the Certificate Server's CRL. The CRL is then published according to a set schedule, which, by default, is configured by default as once a week. When a server or an application receives a certificate from a client, it checks the CRL to insure that the certificate has not been revoked. After the CRL is published, the servers and clients must update their CRLs to make sure they are current. Every CRL is assigned a validity period, which means that as long as a CRL is considered valid by a server or application, it does not try to update the CRL. In Certificate Server, the validity period is set at 10% longer than the schedule for publishing the CRL. The extra time is configured to allow for Active Directory replication time, because the CRL can be published to Active Directory.

To manage certificate revocation and CRL publishing, use the following procedure:

1. Open the Certificate Authority and expand the Issued Certificates container.
2. Right-click the certificate that you want to revoke and select **All Tasks/Revoke Certificate**.
3. Select the reason why the certificate is being revoked and click **OK**.
4. To configure the time it takes to publish the CRL, right-click **Revoked Certificates** and choose **Properties**. The publishing interval and the next scheduled update are listed and can be changed on the CRL Publishing Parameters tab.
5. To force an immediate publication of the CRL, right-click **Revoked Certificates** and select **All Tasks/Publish**. Forcing an immediate publication does not affect the publication schedule.

Managing Group Policy Settings for a CA

You can automate many of the processes involved in managing Certificate Services by configuring group policy settings. The Public Key Policies can be configured for either the computer or user. In either case, the configuration options are located under **Windows Settings\Security Settings\Public Key Policies**. See Figure 5-16.

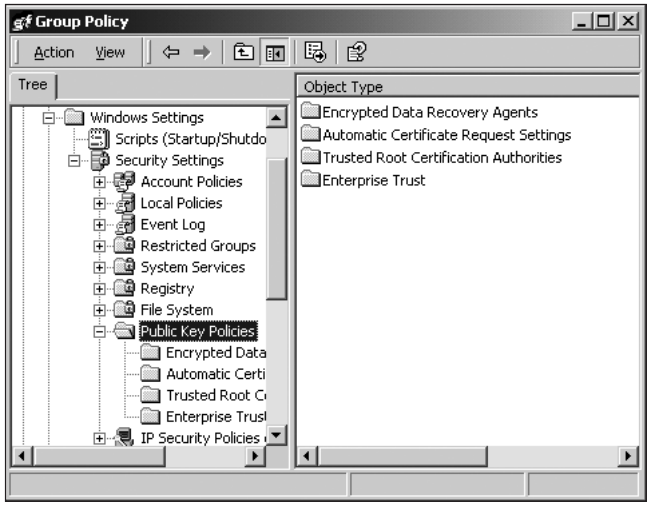

Figure 5-16 The Public Key Policies section in Group Policies

The settings that can be configured include the following:

- **Encrypted data recovery agents**—Used to designate which users are issued a data recovery certificate, enabling them to recover data that has been encrypted using EFS.
- **Automatic Certificate Request Settings**—Used to create certificates automatically for users and computers based on the criteria you select. For example, you may want all users to have a certificate for secure e-mail and web access issued to them. To create a new rule to grant certificates, right-click Automatic Certificate Recovery Agents, and select New\Automatic Certificate Request. The Automatic Certificate Request Setup Wizard starts. You can then choose what types of certificates you want to approve automatically, as well as which CA will be used to grant the certificates.
- **Trusted Root Certification Authorities policy**—Used to configure root certificates to create a trust path to another root CA. To add an additional root CA, you must first obtain a certificate from that root CA, and then import the certificate into Active Directory. All of the clients will then include the CA as part of the trusted root authorities.
- **Enterprise Trust policy**—Used to distribute trust information to clients who list which external CAs can be trusted.

Since group policies can be assigned at an OU level, you have a great deal of flexibility in configuring how the Public Key Policies will be implemented. For example, if you have a group of users that should automatically be granted certificates when they request them, you can put all the users in an OU and configure the group policy for that OU. If you have a group of users that is working closely with another organization, and you want to include the other organization's root CA in the trust path for those users only, you also use group policies at the OU level to implement this option.

Mapping User Accounts to Certificates

One of the main reasons for using certificates is to allow users who may not have an account in Active Directory to have limited access to resources on the Windows 2000 network. However, in Windows 2000, all ACLs are limited to granting permissions based on user and group accounts, and there is no option to assign permissions based solely on certificates. Only security principals can be granted access to the resources in the Windows 2000 domain. However, if you want to provide access to resources for users that have certificates, but not Active Directory user accounts, you can map a certificate to a user account, and then use the accounts to assign permissions.

There are three different ways that certificates can be mapped to user accounts:

- **User principal name mapping**—When an enterprise CA is used, each certificate is mapped to the universal principal name (UPN) of the user who was granted the certificate. When the certificate is used to access a resource

such as a secure Web site, the UPN is used to identify the user in Active Directory, and then the correct permissions are granted.

- **One-to-one mapping**—In this case, each certificate issued by the CA is mapped to a single Windows 2000 user account. One-to-one mapping is a good solution if you are using a standalone CA, and you want remote employees of the company to access secure resources. Another trusted CA could also issue the certificate. By using one-to-one mapping, you can control the level of access for each user.
- **Many-to-one mapping**—In this case, many certificates are mapped to one Active Directory account name. For example, you may be working with another company, and the employees of the company may need access to a secure website. However, you do not want to create a user account for all users from the other company, and you do not want to configure any trusts. You can create a many-to-one mapping where all of the certificates granted by the other company's CA will be mapped to a single Active Directory account, and then you can assign permissions based on that one account.

You can map certificates to user accounts in either IIS or Active Directory. To map certificates to user accounts in Active Directory, use the following procedure.

1. Click **Active Directory Users and Computers** and locate the account to which you want to map the certificates.
2. Right-click the user account and choose **Name Mappings**. See Figure 5-17. (If Name Mappings is not available, make sure that you have selected Advanced Features under the View menu.)

Figure 5-17 The Name Mapppings dialog box

3. Click **Add** and enter the path for the certificate you want to map to this account. You will need to locate the certificate file (.cer extension) and open the file. You are then presented with a choice of what type of mapping you want to create. See Figure 5-18.

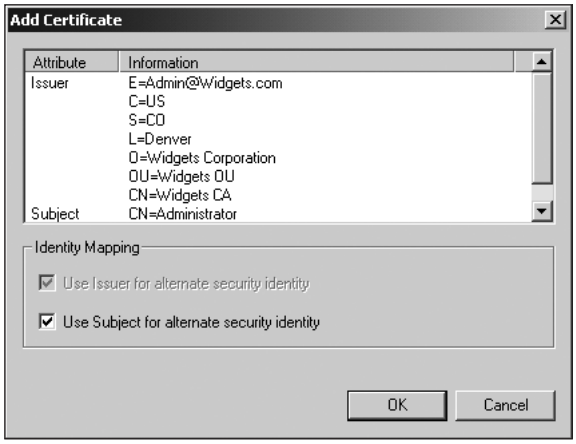

Figure 5-18 Choosing the Security Identity mapping

4. If you are mapping the certificate to a single account, make sure that **Use Subject for alternate security identity** is selected.
5. If you are mapping the certificate to multiple users, and you want to map the account based on the subject (the user name) of the certificate, regardless of the issuer, then make sure that only **Use Subject for alternate security identity** is selected.
6. If you want to map any certificate issued by a particular CA to this account, insure that only **Use Issuer for alternate security identity** is selected.

Certificate Server Client Implementation

All of the administrative tasks discussed up to this point have been tasks performed on the servers by network administrators. The only component of implementing PKI that requires the involvement of the network clients is requesting and installing new certificates. Windows 2000 Certificate Server provides three ways to apply for and manage client certificates:

- Automatically through group policy settings.
- Using the Certificates snap-in. This tool can be used only to request and renew certificates from an enterprise CA.
- Using the Certificate Services Web page *http://win2k-dc1/certsrv/*

In most cases, the easiest way for clients to obtain and install certificates is to use the Certificate Services web page. This virtual directory is installed automatically on the Windows 2000 Certificate Server. To request a new certificate using the Certificate Services Web page, use the following procedure:

1. Open the **Certificate Services** Web page on the Certificate Server *http://servername/certsrv/*. See Figure 5-19.

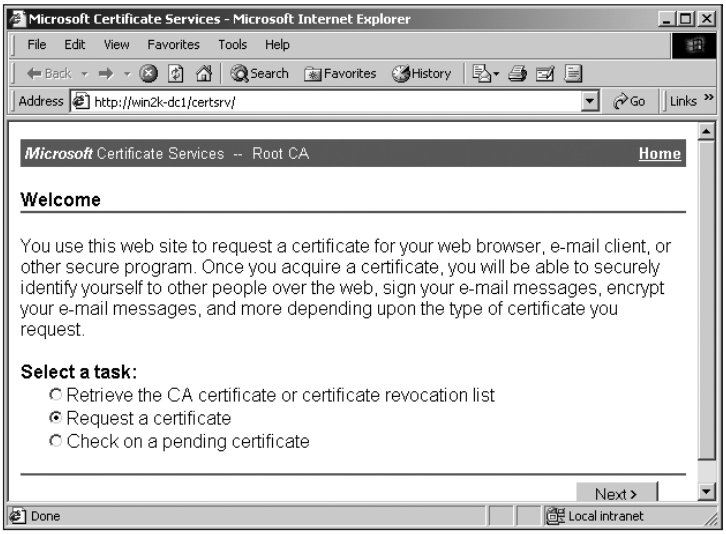

Figure 5-19 Requesting a certificate from a Certificate Server

2. Accept the default task to **Request a certificate** and then click **Next**.
3. Choose the type of certificate that you are requesting. If you select **Advanced Request**, you are given additional options such as key size and choosing other types of certificates.
4. Fill in your personal information. If you are using an enterprise CA, this information is extracted automatically from Active Directory.
5. Click **More Options** if you want to choose a non-default cryptographic service provider, which is the service that creates the public and private keys. As well, you can apply for certificates other than user certificates. See Figure 5-20.

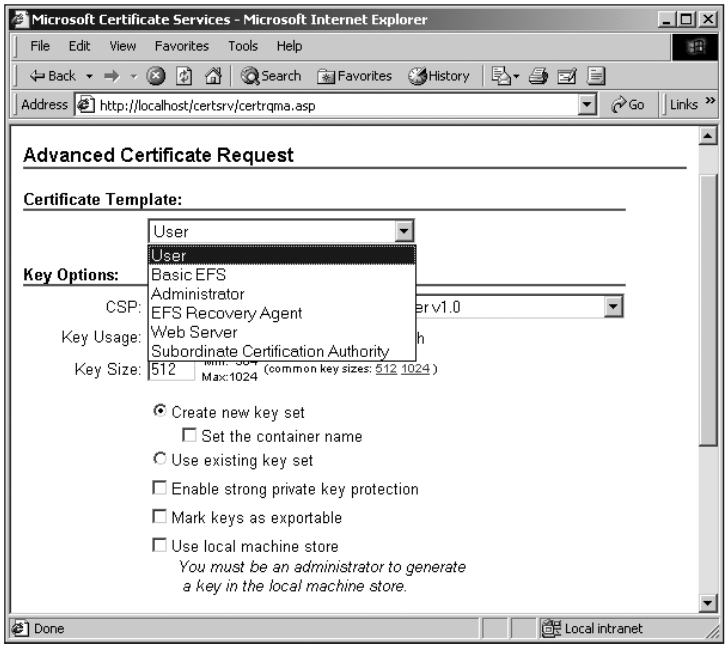

Figure 5-20 Configuring Advanced Certificate Request settings

6. Click **Submit**.

If you are using an enterprise CA, the request is automatically processed and you receive a response immediately as shown in Figure 5-21. If you are using a standalone CA, you receive a message indicating that the administrator must approve your certificate.

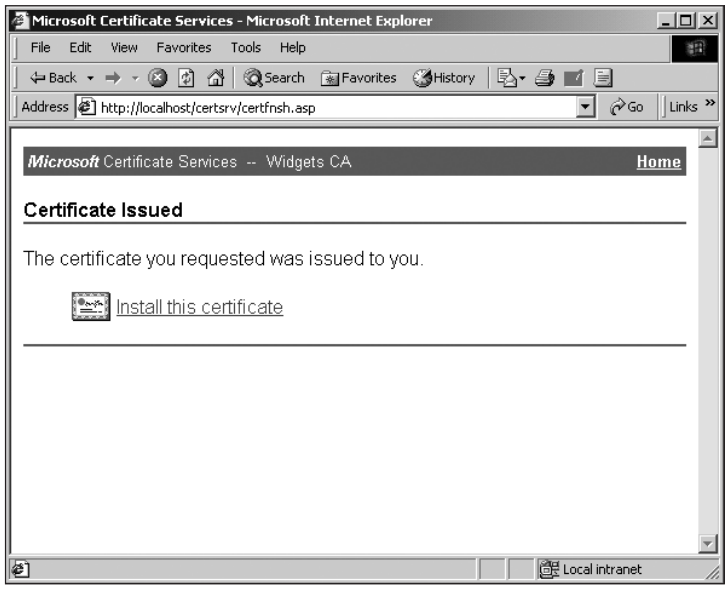

Figure 5-21 Certificate Server response to the client request

After the client has made the request, the user can view the status of the request from the Certificate Server Web page. Once the certificate has been approved, the client can retrieve the certificate using the Web site and install the certificate on the computer.

The Certificates snap-in is also used to request and manage certificates. To use the Certificates snap-in, create a custom MMC and add the Certificates snap-in. The interface is shown in Figure 5-22. Using this snap-in, you can apply for new certificates from an enterprise CA, or manage the certificates that are already installed on the computer. For example, you can use this interface to renew a certificate and export it to a file for backup, or to install the certificate on another computer.

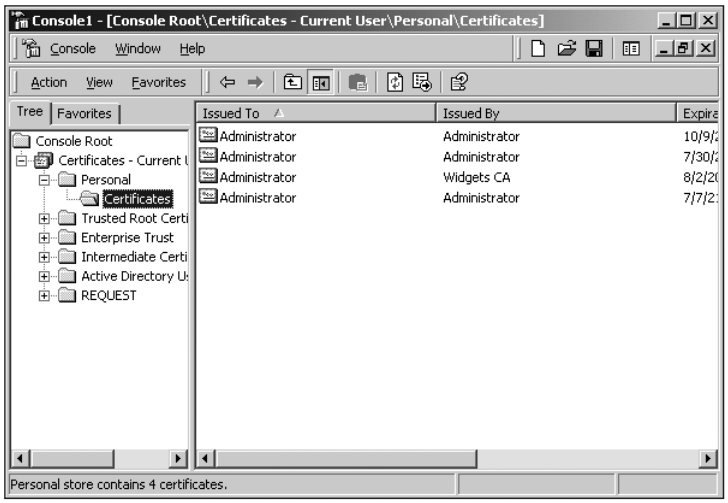

Figure 5-22 The Certificates snap-in

Integration with Third Party CAs

The most important limitation when using the Windows 2000 Certificate Server is the management of the trust path, if you have many customers outside the organization. When you install a root CA, it does not know about the existence of any CAs above it in a trust path, so the certificate trust path stops at the root CA. If all your certificate users are internal to your organization, this is not a difficult problem to overcome. All you have to do is use an Enterprise Root CA, and the Root CA will be included automatically in the trust path for all users who get a certificate from the server or any other server in the CA hierarchy. However, if most of your certificate users are outside your organization, then the management of the trust path is much more difficult because there is no automatic way to include your CA in the trust path for all the users. If your CA is not included in the trust path for the clients, then the users will either not be able to access any resources that require certificates, or they will get an error message every time they connect.

Dealing with the trust path issue is much easier if you use a third-party commercial CA. Most Web-enabled applications like web browsers are already configured to trust a wide range of commercial CAs. Figure 5-23 shows a small part of the Trusted Root Certification Authorities list configured by default in Internet Explorer. If you use a commercial CA certificate to secure your Web site, the web browser is already configured with a trust path to your CA, so the client connection is seamless.

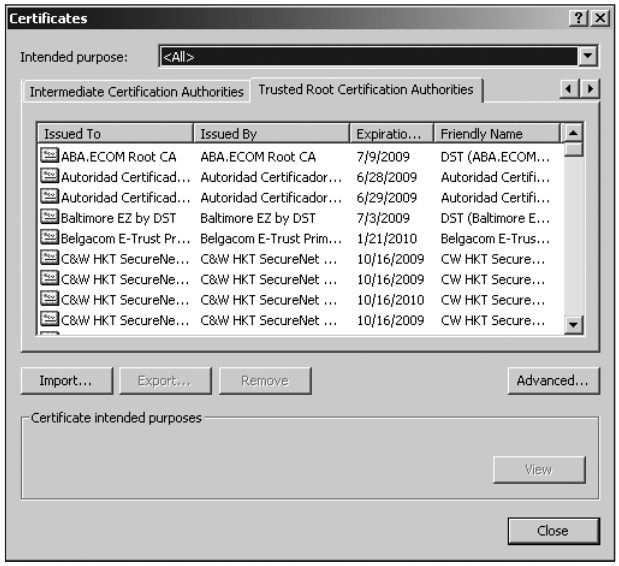

Figure 5-23 The Trusted Root Certification Authorities

Essentially, there are two ways that you can use third-party CAs: you can use only third-party CAs, or you can integrate your internal Windows 2000 CA hierarchy with a third-party CA.

Choosing a Third-Party PKI Solution

If most of your certificate users are outside of your organization, you should choose a third-party PKI solution. In addition to making the management of the trust path easier, the third-party PKI solution has other advantages including the following:

- You do not have to manage your own PKI. The management of a large PKI requires a large amount of administrative effort. By choosing a third-party CA, you are delegating those tasks to a service provider dedicated to that function.
- In addition to decreasing your work load by delegating the administration of the PKI, you can also take advantage of the expertise and experience of the commercial CA. Managing a CA involves many administrative tasks and security issues that are unique to the PKI environment, so you can use the CA's dedicated expertise, rather than developing the expertise yourself.
- A greater level of client trust in your certificate will result. Most clients will be more comfortable trusting a certificate issued by a prominent third-party CA than they would be with a certificate issued by your CA.

The most important disadvantage of choosing a third-party CA is that you might have to give up some administrative and security-related flexibility. If you choose a third-party

CA, you will have to fit your requirements into the services that the CA provides, and you may not be able to implement all the options that you want. The second disadvantage is cost—most commercial CAs charge a per-certificate fee for managing the certificates, and the cost can be significant in a large organization.

Integrating Windows 2000 PKI and a Third-Party PKI

The second option for integrating with a third-party PKI is to implement an internal Certificate Server CA hierarchy that is part of a larger commercial CA hierarchy. To implement this option, install a Windows 2000 subordinate standalone CA, and send the subordinate certificate request to a third-party CA. The commercial CA will investigate your company to determine whether you are a trustworthy recipient of the subordinate CA certificate. If your company is approved, you will receive the subordinate CA certificate. Since your certificates now include the trust path to the commercial root CA, your certificates will be accepted automatically by clients.

If you are planning on integrating with a third-party CA in this way, you need to confirm that all the components in the integration are compatible. The current PKI infrastructure is based on a series of open standards that everyone should be following. For example, the current certificate format is X.509 Version 3, with most PKIs also supporting v1. Everyone uses essentially the same encryption algorithms. SSL is a widely accepted standard. However, many companies deviate slightly from the standards (usually called "enhancements" in the sales literature), and these slight deviations can result in interoperability difficulties. If you are integrating your internal CA with an external CA, you need to thoroughly test the integration to insure that all the required functionality is available.

Planning Best Practices

- Even though many of the administrative tasks involved in implementing a PKI can be automated using an enterprise CA, you will still have to spend a significant amount of administrative effort maintaining your PKI. In particular, managing certificate revocation and recovering lost private keys require considerable time.
- One of the crucial components of implementing a PKI is the management of the client certificates and private keys. By default, these components are stored on the client hard disk and would be lost if the hard disk fails, or if it is formatted. Teaching users how to create backup copies of their certificates and private keys is essential in implementing a PKI solution.
- If you are using certificates to secure a Web site that is accessible on the Internet, your potential clients could include people from anywhere in the world. If this is the case, you should always use a commercial CA certificate to reassure the clients that your Web site is secure.

- Windows 2000 Certificate Server automatically creates an audit trail for all the certificates that are managed by the server. The auditing includes all issued and revoked certificates, as well as all pending requests and all failed certificate requests. This audit trail can be viewed with the Certificate Authority snap-in. An essential component to managing the Certificate Server will be monitoring the audit trail.
- A Certificate Server cannot assign any certificate that expires after the Certificate Server certificate. For example, if you install a certificate that is valid for two years on a subordinate CA, the CA cannot issue certificates with more than a two-year expiration. After 18 months, the CA will be able to assign certificates that are only good for six months. To manage this issue, make sure that the certificates issued to CAs are valid for longer periods of time than the client certificates, and maintain a regular renewal cycle on the CAs.

Chapter Summary

- Public Key Infrastructure is used to provide distributed security so that authentication, encryption, and digital signature services can be extended beyond Active Directory. Instead of using a shared secret to authenticate users, PKI uses certificates.
- The essential components of PKI are certificates, public and private keys, and certificate authorities that assign the certificates. CAs are usually organized in a hierarchical structure, and all of the CAs in that hierarchy share a common trust path to the root CA.
- If you are planning to implement a PKI as part of your security plan, then you need to plan your PKI implementation carefully. The elements of your PKI planning include designing the Certificate Server hierarchy, planning the Certificate Server type, identifying client certificate needs, defining certificate policies, and planning for certificate revocation.
- Windows 2000 Certificate Server provides a full-featured PKI solution. Implementing the PKI using Certificate Server includes installing the root and subordinate CA servers, configuring servers and applications to use certificates, managing user certification requests, managing certification revocations, managing group policy settings to manage certificates, and mapping user accounts to certificates.
- In some cases you may want to integrate your Windows 2000 environment with a third-party PKI solution. You have two options to configure this: use only third-party CAs, or implement your Windows 2000 CA hierarchy as part of a commercial CA hierarchy.

KEY TERMS

Certificate Authority (CA) — A certificate server that assigns certificates to other certificate servers or clients.

certificate revocation lists (CRLs) — Lists maintained by CAs of certificates that have been revoked. Clients and servers should check the CRL before granting access based on a certificate.

Certificate Server — Microsoft's standards-based PKI service available for Windows 2000.

client authentication — Used to insure the authenticity of the client by checking the validity and authenticity of the client's certificate.

digital certificates — A digital entity assigned to a user or computer that is used to vouch for the identity of the certificate holder.

digital signature — A mechanism used to insure the identity of the sender of a message and also to insure the integrity of the message.

enterprise CA — An implementation of Windows 2000 Certificate Server that requires Active Directory and is completely integrated with Active Directory. This integration can simplify the administration of certificates, because you can configure policies that automate the process of granting, renewing, and revoking certificates.

hierarchical CA structure — A configuration of certificate authorities into a hierarchical structure where each CA issues the certificate for the CA underneath it in the hierarchy. All CAs in the same hierarchy share the same root CA.

IP Security (IPSec) — An Internet Engineering Task Force (IETF) solution that can use PKI to protect data on the network. With IPSec, all data transmitted on a network can be encrypted.

message digest — The result of a mathematical hash being applied to a message. The message digest is used as part of a digital signature to insure that the message was not tampered with while it was transmitted on the network.

Pretty Good Privacy (PGP) — A service that encrypts and signs e-mail between two users using the same third-party software.

private key — A key that is part of a certificate and is known only to the user who holds the certificate. It can be stored on the computer's hard drive, or as part of a roaming profile, or on a different device, such as a smart card.

public key infrastructure (PKI) — A security model that uses certificates and private and public keys to authenticate users and computers and to encrypt and digitally sign data.

public key — A key that is made available to anyone who asks for it. The public key is attached to the certificate.

root CA — The top server in a CA hierarchy. The root CA issues its own root certificate.

Secure Sockets Layer (SSL) and **Transport Layer Security (TLS)** — Two open standards that require PKI to authenticate and encrypt data flowing between Web servers and Web clients.

Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3 — A proposed standard for secure e-mail where e-mail can be encrypted and digitally signed as it is sent between two users.

server authentication — The process of ensuring that a server is authentic and not an imposter. The authentication in this case is certificate-based; if the server can prove that it has a valid certificate that the client trusts, then the client assumes that the server is authentic.

session key (also called **bulk encryption key** or **symmetric key)** — The key that is used to encrypt a message when it is sent across the network. Both the sending and receiving computers use the same session key, which is encrypted and decrypted using the private and public keys.

SSL handshake — The process of authenticating servers and clients, and creating a session key that can be used by SSL to encrypt all data.

standalone CA — An implementation of Windows 2000 Certificate Server that does not require Active Directory and can be integrated with third-party CAs.

subordinate CA — A CA that is at a lower level in the CA hierarchy. Subordinate CA certificates are always issued by the CA immediately above the subordinate CA in the CA hierarchy.

trust path — A mechanism for computers with certificates from different CAs to trust each other's certificates. If the two computer certificates have the same root CA, then they share a trust path and will trust each other's certificates.

X.509 Version 3 — The current standard for PKI certificates. The certificate includes information about the person, computer, or service to which the certificate has been issued, information about the certificate itself, and information about the Certificate Authority that issued the certificate.

Review Questions

1. When sending an encrypted e-mail message to a friend, which key will be used to encrypt the message?
   a. your public key
   b. your private key
   c. your friend's public key
   d. your friend's private key
2. Your supervisor wants a method to send information across the Internet in a manner that guarantees that the information has not been modified in transit. Which PKI component provides this functionality?
   a. Private Key encryption
   b. Public Key encryption

c. digital signatures
d. certificates
e. Enterprise Certificate Authorities

3. What type of security is best for protecting information on a stolen laptop computer?
   a. authentication
   b. access control lists
   c. IPSEC
   e. Encrypting File System (EFS)
   f. digital signatures
4. Which PKI component grants digital certificates?
   a. Ticket Granting Service (TGS)
   b. Certificate Authority (CA)
   c. Authentication Server (AS)
   d. Domain Controller
   e. Key Distribution Center (KDC)
5. Which protocol allows users to securely log on to a Web server using their domain username and password regardless of the Web browser they are using?
   a. Kerberos Version 5
   b. certificate-based authentication
   c. digest authentication
   d. SSL
   e. RADIUS
6. You have installed an Enterprise Certificate Authority on your network and then used a certificate issued by this CA to configure SSL on your Web server. When your clients access the web site, they are told that the certificate is issued by an untrusted CA. Why?
   a. The clients are configured incorrectly.
   b. You did not configure the Web server to trust your CA.
   c. Your CA is not included in any trust path for the client.
   d. Your CA is not trustworthy.
7. A session key is used to:
   a. encrypt the public key when sending an encrypted message
   b. digitally sign messages
   c. authenticate a destination computer
   d. improve the performance of a computer involved in sending encrypted information

8. Which key does the receiver of a digitally signed message use to confirm that the message has not been modified in transit?
   a. the receiver's public key
   b. the receiver's private key
   c. the sender's public key
   d. the sender's private key
9. You would like your PKI solution to be integrated with Active Directory. What type of CA must you install first?
   a. enterprise root CA
   b. enterprise subordinate CA
   c. standalone root CA
   d. standalone subordinate CA
   e. none of the above
10. What would you have to do to rebuild your PKI if an attacker managed to gain access to your root CA and export all of the certificates on the server?
    a. rebuild the entire PKI with a new Root CA
    b. restore the root CA from a backup tape
    c. re-issue certificates to all subordinate CAs
    d. add the root CA certificate to the Certificate Revocation List
11. You are an administrator in a large company that is using PKI. It has been decided that you will have a CA within your region and you are responsible for installing it. You have been told your PKI system is not integrated with Active Directory but you will need to apply to a CA at head office to obtain a certificate. Which type of CA will you install?
    a. enterprise root CA
    b. enterprise subordinate CA
    c. standalone root CA
    d. standalone subordinate CA
    e. none of the above
12. In order to have a completely secure Web site, you need to configure:
    a. your own Root CA
    b. server authentication
    c. client authentication
    d. confidentiality

13. The benefits of having subordinate CAs in your PKI hierarchy include which of the following?
    a. making the PKI more secure
    b. providing redundancy in the case of a root CA failure
    c. delegating the tasks of assigning certificates
    d. providing load balancing
14. S/MIME is:
    a. a standard used to ensure the security of all traffic to and from a Web site
    b. a standard used to ensure the security of all network traffic
    c. a standard used to ensure the security of e-mail messages
    d. a standard used to configure the trust paths for Web clients
15. When you install and configure an enterprise root CA:
    a. all of certificate requests made to the server will automatically be granted.
    b. you must already have Active Directory installed.
    c. you can control which certificates will be granted through Active Directory.
    d. you cannot install Active Directory on the same computer.
16. Which certificate template would you use to set up SSL for intranet use?
    a. Domain Controller
    b. Web server
    c. computer
    d. user
    e. authenticated session
17. You have installed an enterprise root CA and an IIS server. You used the Web Server Certificate Request to request and install a certificate from the CA. Everything seems to work, but you notice that clients can still connect to the server using just HTTP rather than HTTPS. What step did you forget?
    a. to create a trust path to an external CA
    b. to configure the Web site to require secure channels
    c. to configure the clients to require secure channels
    d. to configure the Web site to require client certificates
18. You have just received a new certificate from your internal CA and you notice that the expiration date for the certificate is only two months from now. You are sure that the server is configured to hand out certificates that are valid for two years. What would you check first?
    a. the expiration date on the CA certificate
    b. the CRL
    c. the date on the client machine to make sure that it is accurate
    d. whether the server has been rebooted in the last two months

19. You would like to configure a CA as a subordinate CA to a third-party CA. What must you do to accomplish this?
    a. import the public key of the third-party CA
    b. import the private key of the third-party CA
    c. import the trusted root certificate of the third-party CA
    d. export your trusted root certificate and give it to the third-party CA
20. You would like to give all of the users in a partner company controlled access to some parts of your Web site, but you do not want to create user accounts for all the users. You know that the partner company has a PKI infrastructure in place. How could you set this up?
    a. issue a certificate from your CA for all the users at the partner organization
    b. create a one-to-one mapping of certificates from your partner company's CA to user accounts
    c. create a many-to-one mapping of certificates from your partner company's CA to a user account
    d. you can't do this

HANDS-ON PROJECTS

Project 5-1

In this hands-on project, you will install Certificate Services.

To install Certificate Services on the server:

1. Log on to your Windows 2000 computer as an administrator.
2. Click **Start**, point to **Settings**, and click **Control Panel**.
3. Double-click **Add/Remove Programs**.
4. Click **Add/Remove Windows Components**.
5. Enable the **Certificate Services** checkbox. Click **Yes** to continue.
6. Click **Next** to accept running Terminal Services in Remote Administration Mode.
7. Click **Enterprise root CA**. Click **Next**.

8. Enter the required company information as follows:

   CA Name: **Lonestar Publishing Root CA**

   Organization: **Lonestar Publishing**

   Organizational Unit: **Head Office**

   City: **Winnipeg**

   State/Province: **MB**

   Country/Region: **CA**

   E-mail: **Admin@Lonestar.com**

   Click **Next**.

9. Click **Next** to accept the default locations for the certificate database and the certificate database log.
10. Click **OK** to stop IIS if it running. If prompted, insert the Windows 2000 Server CD and click **OK**.
11. Click **Finish** and close all windows.

Project 5-2

In this hands-on project, you will configure Certificate Services to allow users to use certificates assigned to smart cards.

To configure a new Certificate Template:

1. Be sure to be logged on as the administrator.
2. Click **Start**, point to **Programs**, point to **Administrative Tools**, and click **Certification Authority**.
3. In the left pane, expand the CA that you created in project 5-1.
4. Click **Policy Settings**. Note the certificate templates that are already installed by default.
5. Add a new certificate template to your CA to allow for the creation of smart cards by right-clicking on **Policy Settings**.
6. Point to **New**, and click **Certificate to Issue**.
7. Click **Smart card Logon**. Click **OK**.
8. Close all windows.

Project 5-3

In this hands-on activity, you will create a User Certificate for the administrator to use to encrypt e-mail.

To enroll for a new certificate:

1. Be sure to be logged on as the administrator.
2. Open **Internet Explorer**.
3. Go to the URL *http://127.0.0.1/certsrv*.
4. Log in as **administrator** with your domain password.
5. Select **Request a certificate**. Click **Next**.
6. Select **User certificate request**. Click **Next**.
7. Click **Submit** to process the request.
8. Click **Install this certificate** to install it in Windows. Note that it has been successfully installed.
9. Close **Internet Explorer**.

Project 5-4

In this hands-on project, you will revoke the certificate that was issued in project 5-3.

To revoke a certificate:

1. Be sure to be logged on as the administrator.
2. Click **Start**, point to **Programs**, point to **Administrative Tools**, and click **Certification Authority**.
3. In the left pane, expand the root CA.
4. Click **Issued Certificates**. Notice the issued certificate in the rightmost details pane.
5. Right-click the issued certificate and choose **All Tasks**, **Revoke Certificate**. Be sure to revoke the certificate that has the same effective date and time as that issued in Project 5-3.
6. At the **Certificate Revocation** dialog box, choose **Change of Affiliation** for the reason code.
7. Click **Revoked Certificates**. Notice the revoked certificate indicated with the red X.
8. Close all windows and log off.

CASE PROJECTS

Case Project 5-1

1. The management at Southdale Property Management has asked you to investigate the need for a PKI for the company. Based on the information you have collected so far, does Southdale Property Management have a need for a PKI? If there is a need for a PKI, what would be the best way to implement it?

Case Project 5-2

2. Fleetwood Credit Union implemented a PKI solution when it set up the secure Web site for its clients two years ago. The Web server was configured with a server certificate from a third-party CA and then configured to require secure client connections, but not to require 128-bit encryption or client certificates. The management team has asked you to evaluate the solution and provide recommendations in three areas:

   - Should they configure an internal Windows 2000 Certificate Server and use it to issue the certificate for the Web site? This would save over $1000 per year because they would not require the commercial certificate.
   - Should they require 128-bit encryption for all clients?
   - Should they require client certificates?